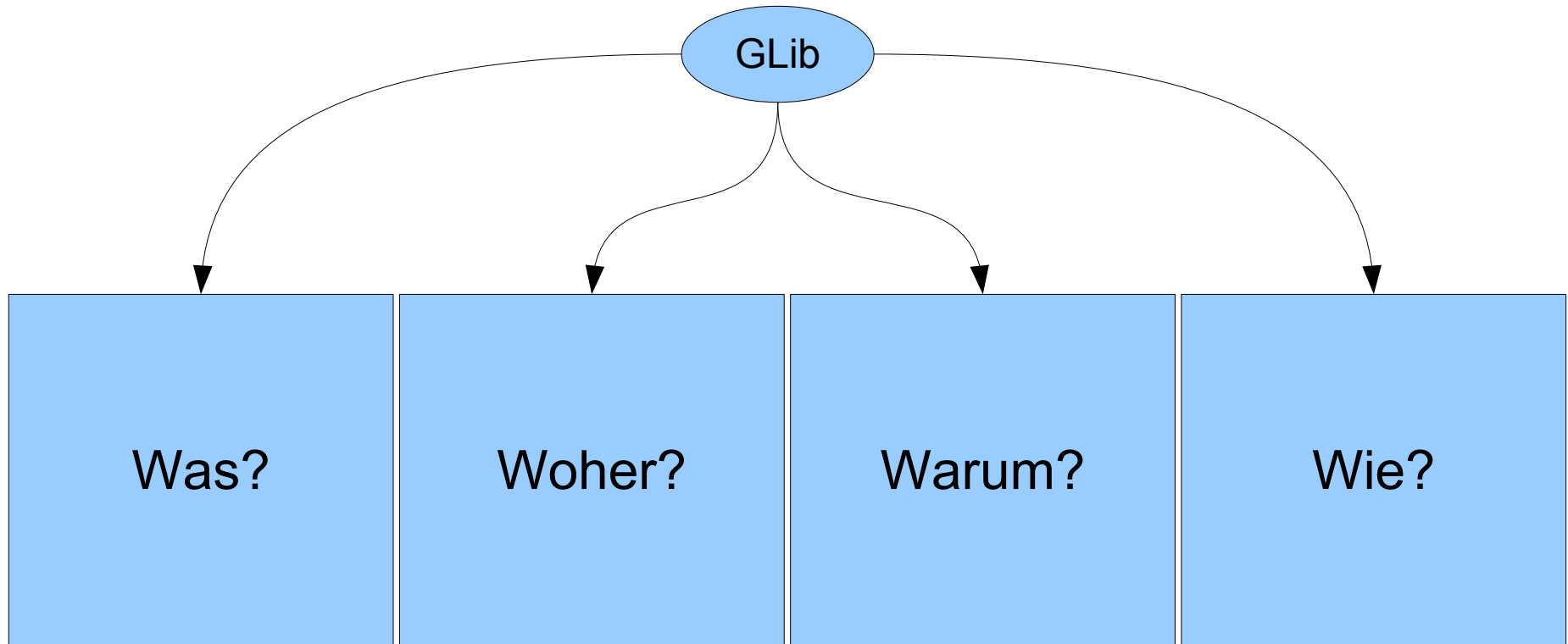


Gnome Library - GLib

Esteban Böhmecke

Agenda



Was ist die GLib?

- Eine in C geschriebene Bibliothek
- Verfügbar unter:
 - ◆ Unix
 - ◆ Windows
 - ◆ OS/2
 - ◆ BeOS
- Lizenz: LGPL (freie Software)

Was ist die GLib?

- Viele verschiedene Funktionen:
 - ◆ Basistypen
 - ◆ Komplexe Datenstrukturen (z.B: verkettete Listen)
 - ◆ Zeichenketten (Strings)
 - ◆ Felder (Arrays)
 - ◆ Threads
 - ◆ Zeitfunktionen (arbeiten mit Timer)

Woher kommt die GLib?

- Entwickelt aus GTK+
 - (Bibliothek zur GUI Erstellung)
- Erste Versionen von GTK+ erhielten schon:
 - ◆ Verkettete Listen
 - ◆ Binärbäume
 - ◆ „Mitwachsende“ Strings
 - ◆ System zur Objektorientieren Programmierung

Woher kommt die GLib?

- Viele GTK+ Funktionen ausgelagert:
 - GLib Bibliothek
- Auslagerung vom Objektorientierte System
 - GObject-Bibliothek
- Auslagerung vom systemnahen Zeichenmethoden
 - GTK+ plattformübergreifend benutzbar

Warum sollte man die GLib benutzen?

- Viele Hilfsmittel die uns das Leben erleichtern

GString

GThread

GAsyncQueue

GArray

GDateTime

Basistypen

GError

String Utility Functions

Random Numbers

GTimer

GHashTable

GTimeZone

GAllocator

Wie benutzt man die GLib?

GString

- `g_string_append (GString *string, const gchar *str)`

```
1. GString *testString;  
2.  
3. testString= g_string_new("Pro");  
4.  
5. g_string_append (testString, "seminar");
```

- `g_string_prepend (GString *string, const gchar *str)`

```
1. GString *testString;  
2.  
3. testString= g_string_new("seminar");  
4.  
5. g_string_prepend (testString, "Pro");
```


Wie benutzt man die GLib?

GString

- `g_string_insert (GString *gstring, gssize pos, const gchar *str)`

```
1. GString *testString;  
2.  
3. testString = g_string_new("Pronar");  
4.  
5. g_string_insert(testString, 2, "semi");
```

- `g_string_truncate (GString *gstring, gint len)`

```
1. GString *testString;  
2.  
3. testString = g_string_new("ProseminarTest");  
4.  
5. g_string_truncate(testString, 10);
```

Wie benutzt man die GLib?

GString

- Truncate Methode in der C Standard-Bibliothek:

```
1. #include <string.h>
2.
3. size_t truncateCharArray(size_t maxLength, char *buffer)
4. {
5.     /* NULL string oder ungültige Länge? */
6.     if (buffer == NULL || !maxLength)
7.         return 0;
8.
9.     /* string zerschneiden; gibt länge vom zerschnittenen
10.        string zurück oder 0 wenn ein Fehler auftritt */
11.
12.     if (maxLength <= strlen(buffer))
13.     {
14.         buffer[maxLength] = '\0';
15.         return maxLength;
16.     }
17.     else
18.         return 0;
19. }
```

Quelle: [1]

Wie benutzt man die GLib?

GString

- Viele andere String bearbeitende Methoden:
 - ♦ `g_string_assign (GString *string, const gchar *rval);`
 - ♦ `g_string_overwrite (GString *string, gsize pos, const gchar *val);`
 - ♦ `g_string_erase (GString *string, gssize pos, gssize len);`
 - ♦ `g_string_hash (const GString *string);`

Wie benutzt man die GLib?

String Utility Functions

- Automatisch beim „glib.h“-Include vorhanden
- `g_string_ascii_up (GString *string);`
- `g_string_ascii_down (GString *string);`
- `g_ascii_isdigit (gchar c);`
- `g_str_has_prefix (const gchar *str, const gchar *prefix);`
- `g_str_has_suffix (const gchar *str, const gchar *suffix);`

Wie benutzt man die GLib?

String Utility Functions

- `g_str_has_suffix (const gchar *str, const gchar *suffix);`

```
1. gboolean suffix_value;
2.
3. suffix_value = g_str_has_suffix("ftp://www.test.com/test-1.0-1.tar.gz", ".tar.gz");
4.
5. if(suffix_value == TRUE)
6. {
7.     g_print("Suffix is true \n");
8. }
9. else {
10.     g_print("Suffix is false \n");
11. }
```

Quelle: [2]

- `g_str_has_prefix()` funktioniert sehr ähnlich

Wie benutzt man die GLib?

GArray

- `g_array_new(...);`
 - ◆ `gboolean zero_terminated`
 - ◆ `gboolean clear_`
 - ◆ `guint element_size`
- `g_array_append_val(a, v);`
 - ◆ A: ein GArray.
 - ◆ V: wert der an GArray dran gehängt werden soll.
- `g_array_prepend_val(a,v);`

Wie benutzt man die GLib?

GArray

- `g_array_append_val(a, v)`

```
1. #include <glib.h>
2.
3. GArray* meinArray = g_array_new(FALSE, FALSE, sizeof(char*));
4.
5. char* erster = "hallo", *zweiter = "test", *dritter = "welt";
6.
7. g_array_append_val(meinArray, erster);
8. g_array_append_val(meinArray, zweiter);
9. g_array_append_val(meinArray, dritter);
10.
11. g_array_remove_index(meinArray, 1);
12.
13. g_array_free(meinArray, FALSE);
```

Wie benutzt man die GLib?

GHashTable

- `g_hash_table_new (...);`
 - ♦ `GHashFunc hash_func`: (z.B: `g_int_hash()`, `g_str_hash()`)
 - ♦ `GEqualFunc key_equal_func`: (z.B: `g_int_equal()`, and `g_str_equal()`)
- `g_hash_table_insert (...);`
 - ♦ `GHashTable *hash_table`
 - ♦ `Gpointer key`
 - ♦ `Gpointer value`
- `g_hash_table_lookup (GHashTable *hashtable, gconstpointer key);`

Wie benutzt man die GLib?

GTimer

- `g_timer_elapsed (...);`
 - ♦ `GTimer *timer,`
 - ♦ `Gulong *microseconds`
 - ♦ Return: `gdouble` (verbrauchte Zeit)

```
1. /* Gibt aus wieviel Zeit seit dem starten des Timers vergangen ist*/
2. GTimer * timer = (GTimer*) data;
3. gulong verbrauchte_microsekunden;
4. gdouble verbrauchte_zeit = g_timer_elapsed (timer, &verbrauchte_microsekunden);
5.
6. /* Sag dem benutzer wieviel Zeit diese Funktion gedauert hat */
7. printf ("Diese Funktion haz %.2f Sekunden gedauert.\n", verbrauchte_zeit);
8.
9. /* Timer vom Speicher freigeben */
10. g_timer_destroy (timer);
```

Zusammenfassung

- Viele hilfreiche Funktionen für Programmierer
- Einfach zu benutzen, gute Dokumentation
- Besonders nützlich für C Anfänger (Javaprogramm.)
- Auch Funktionen für Experten (Threads, ...)

Quellen

- Offizielle GLib Dokumentation: <http://developer.gnome.org/glib/2.28/>
- http://www.mathematik.uni-ulm.de/help/gtk+-1.1.3/gtk_tut-22.html
- <http://de.wikipedia.org/wiki/GLib>
- <http://de.wikipedia.org/wiki/GTK%2B>
- <http://www.linuxjunkies.org/programming/GUI/gnome/glib.html>

- Codebsp. [1]: <http://bytes.com/topic/c/answers/212141-truncating-char-array>
- Codebsp. [2]: <http://www.gtkforums.com/viewtopic.php?t=280>
- Codebsp. [3]: <http://www.ibm.com/developerworks/linux/tutorials/l-glib/section6.html>